

Application No. 10/026,364**Atty Docket No. JGR 1007-1****REMARKS**

Claims 1-44 are pending in this application. Claims 1-44 are rejected under 35 USC § 103. All rejections are traversed.

Applicants note with appreciation the withdrawal of various objections and rejections.

Rejection Under 35 U.S.C. § 103(a) of Claims 1-3, 6-7, 15-16, 24-29

The Examiner rejects **claims 1-3, 6-7, 15-16, 24-29** under 35 U.S.C. § 103(a) as unpatentable over Harold, XML in a Nutshell, January 2001 in view of Yalcinalp (US 6,507,857). We begin with a review of some of the references cited and applied by the Examiner.

Technology Review***Harold Chapter 9 Reference***

To put Harold Chapter 9 in perspective, it is useful to understand what XPath is and how it relates to XSLT. Both of these technologies are acknowledged and discussed in the application. XPath is short for XML Path Language. Industry standards organization W3C published version 1.0 of the XPath specification as a Recommendation on 16 November 1999. It is currently accessible at www.w3.org/TR/xpath. The Introduction to the Recommendation explains:

XPath is the result of an effort to provide a common syntax and semantics for functionality shared between XSL Transformations [XSLT] and XPointer [XPointer]. The primary purpose of XPath is to address parts of an XML [XML] document. ... XPath uses a compact, non-XML syntax to facilitate use of XPath within URLs and XML attribute values. XPath operates on the abstract, logical structure of an XML document, rather than its surface syntax. XPath gets its name from its use of a path notation as in URLs for navigating through the hierarchical structure of an XML document.

In addition to its use for addressing, XPath is also designed so that it has a natural subset that can be used for matching (testing whether or not a node matches a pattern); this use of XPath is described in XSLT.

Id., § 1 (hyperlinks omitted). Harold Chapter 9 is devoted to a language within a language: the XPath language for specification of a path in an XSLT stylesheet. XPath is not an XML manipulation language.

Application No. 10/026,364

Atty Docket No. JGR 1007-1

XSLT is an XML transformation language in which nodes can be identified by an XPath expression. Example 9-3 at Harold pp. 5-6 illustrates applying XSLT to generate an HTML table from an XML document. On p. 15, two more examples are given of transforming a "profession" element of an XML source into output, one performing a conditional transformation when the contents of the XML source has the value "physicist" or "computer scientist" and the other italicizing "*computer scientist*" in the output stream. (It may be helpful to review "Selecting the Current Element with ." on p. 9, which explains italicizing output, to better understand the

`<i><xsl:value-of select="." /></i>` statement on p. 15.) Both of Harold's examples on p. 15 produce an output stream, according to the reference. Neither example updates the XML source.

Yalcinalp Reference

The Yalcinalp reference teaches extending the XSL language and XSLT processors invoke external functions from within a stylesheet, as an approach to document transformation. The word "transformation" is used in the invention's title. To set the stage for external function calls, Yalcinalp introduces the XSL stylesheet language used by XSLT processors:

2. Description of the Related Art

Systems' and applications' use of documents has become so prolific that style sheets are now often used to help manage the documents' display. Style sheets provide greater flexibility and control over the display of a document's content. XSL style sheets also allow the content of documents to be transformed, making them as document transformers where the resulting documents may or may not be used for display. A user may request a document and the application associated with the document will use the information contained in the style sheet to display a new document incorporating the information contained within the style sheet and the requested document.

Id., col. 1. The abstract explains the invention:

When the style sheet processor processes the tags in the style sheet, it recognizes the external component declaration. The style sheet will contain a

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

name of the external component instance and a definition of the method to execute associated with the external component instance, and may contain arguments for the method associated with the external component instance which is executing. The XSLT processor then relinquishes control to the external component to execute the method defined in the style sheet. The results of the method's execution may be placed in the transform document generated by processing the style sheet.

Nothing cited from Yalcinalp suggests updating the XML source, as opposed to transforming it to generate an output stream using an XSLT processor that applies a stylesheet. In fact, the word "update" is not used in Yalcinalp.

More generally, the nature of XSLT processors and XSL stylesheets is to generate something new from queries or "matches" against an XML document.

The Technology Disclosed

This disclosure acknowledges a number of software technologies and extends them. The technologies acknowledged include XML, XPath, XSLT, DOM, SOX and Schematron. These technologies are discussed throughout the application.

Commerce One, the original assignee of these applications, was a pioneer in applying XML technologies to e-commerce. Many fundamental approaches to applying XML to e-commerce and technologies implementing these approaches were first developed by Commerce One's inventors. Their novel combinations of the acknowledged technologies are not found in the references cited.

Applicants urge the Examiner to view this application from the early days of applying XML to e-commerce and to fight against the pervasive influence of hindsight.

Claim 1

Claim 1 includes the limitations:

receiving a character string including one or more sets of:

an update operator;

a path specification identifying a node at which the update operator is to be applied; and

one or more update values;

parsing the character string;

accessing a self-describing, structured document; and

updating said document with the update values at the path specification.

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

These limitations are not found in Harold in view of Yalcinalp.

The Examiner proposes to find all of the elements of claim 1, other than parsing the character string, in Harold Chapter 7.

Receiving a character string has three sub-elements. A character string is a broad limitation, so the Examiner does not mention that he considers an XSL stylesheet to be a character string. We make this explicit for the sake of completeness.

Regarding an update operator, the Examiner reasons, citing page 15, "through the use of Booleans, an update operator, in this case italicization, is applied when criteria are met" and "if the document satisfies the criteria, having the profession element contents equal to 'computer scientist,' then the element is italicized". This reasoning overlooks the difference between updating and transformation. Harold p. 15 describes transformation and creation of an output stream, potentially a new document. Harold p. 15 does not teach updating the XML source. The "match=" condition and the logic to produce italicized output (reproduced *supra*, at 11) are transformation operators, not update operators.

Regarding a path specification, claims 7 and 8 say that XPath may be the path specification used. However, Harold does not, as claimed, use *a path specification identifying a node at which the update operator is to be applied*.

Regarding update values, Harold p. 15 provides a transformation of part of an XML document, not an update value. The XML source is not updated and no update value is intended by Harold.

Regarding accessing a self-describing, structured document, XML is given as an example of a self-describing, structured document at [0042] of the application.

Regarding updating the document accessed, Harold does not update the XML source. Harold uses an XSLT processor to generate a new output stream, a new document.

Regarding parsing, for which the Examiner relies on Yalcinalp, we do not disagree with the assertion that an XSLT processor parses an XSL stylesheet.

Therefore, because Harold has nothing to do with updating or update operators, claim 1 should be allowable over Harold in view of Yalcinalp.

Claims 2-3

Claims 2-3 include the limitations:

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

wherein the character string further includes a document ID

wherein accessing the document includes retrieving the document based on the document ID

These limitations are not found in Harold in view of Yalcinalp. The Examiner proposes to find these limitations in Harold Chapter 7, page 14, paragraph 4, which reads:

General XPath Expressions

So far we've focused on the very useful subset of XPath expressions called location paths. Location paths identify a set of nodes in an XML document and are used in XSLT match patterns and select expressions. However, location paths are not the only possible type of an XPath expression. XPath expressions can also return numbers, booleans, and strings. For instance, all legal XPath expressions include:

We have focused on this paragraph. We do not find a document ID element in an XML document used to select the document. Overall, Harold Chapter 7 is about selecting among elements of a single XML document, not about selecting among XML documents. It is not surprising that Harold does not address these limitations.

Therefore, claims 2 and 3 should be allowable over Harold in view of Yalcinalp.

Claims 6-7

Claims 6-7 should be allowable over Harold in view of Yalcinalp for at least the same reasons as claims 1 and 3, from which they depend.

Claims 15-16

Claims 15-16 include the limitations:

further including accessing an element set list corresponding to a plurality of the update values to be applied at the path specification

These limitations are not found in Harold in view of Yalcinalp. A use of this feature is explained in the specification [0061], "When an alias is used for a set of elements, more than one update value may be associated with a single path specification." While Harold p. 13, ¶ 4 uses an XPath statement to access a group of nodes, the claim language uses *an element set list corresponding to a plurality of the update values to be applied*, not for selecting a group of nodes. One difference is that a series of update values may be ordered, corresponding to an ordering of the element set list, so that several sub-elements can be updated with different values using the element set list to

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

name them (as an alias). In Harold, all of the elements selected by the match statement receive the same treatment.

Therefore, claims 15-16 should be allowable over Harold in view of Yalcinalp.

Claim 24

Claim 24 includes the limitations:

wherein a single update operator applies to a plurality of the sets

These limitations are not found in Harold in view of Yalcinalp. For instance, suppose an organization has a contact record for its web site, including an administrative contact and a technical contact. The *single update operator* may replace the names of the administrative/technical contacts using a replace element operator, with different names for the contact people. In juxtaposition, Harold Chapter 7, p. 15 *italicization* example, does not include an update operator such as "replace element" because Harold discusses transformation, not updating. And, Harold's match statement applies the same transformation into an output stream for each match, it does not apply the same operator, such as add element, delete element or replace element, to different update values in the received character string.

Therefore, claim 24 should be allowable over Harold in view of Yalcinalp.

Claim 25

Claim 25 includes the limitations:

wherein the update operator is implied and not explicit in the character string

These limitations are not found in Harold in view of Yalcinalp.

The Examiner argues that the "@" expression used in match statements (Harold 7) is not an explicit character string, because it includes a wildcard. Reading Harold, it appears that "@" is part of the XPath language for specification of a path. This expression is not an update operator – it cannot be both a path specification expression and an update operator, as claimed.

Therefore, claim 25 should be allowable over Harold in view of Yalcinalp.

Claims 26-29

Claims 26-29 should be allowable over Harold in view of Yalcinalp for at least the same reasons as claim 1, from which they depend. Again, Harold Chapter 7 has

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

nothing to do with updating XML source, such as adding nodes to a document tree, before or after a specific node.

Applicants respectfully submit that claims 1-3, 6-7, 15-16, 24-29 should be allowable over Harold in view of Yalcinalp.

Rejection Under 35 U.S.C. § 103(a) of Claims 4-5, 8 and 17

The Examiner rejects **claims 4-5, 8 and 17** under 35 U.S.C. § 103(a) as unpatentable over Harold, XML in a Nutshell, January 2001 in view of Yalcinalp (US 6,507,857) in further view of Document Object Model (DOM) Tutorial (30 October 2000).

Claim 4

Claim 4 includes the limitations:

wherein a document ID is implied by prior state information

These limitations are not found in Harold in view of Yalcinalp in further view of DOM Tutorial.

Combination of DOM and XSLT is natural, even explained in this application at [0052], in the context of applying a stylesheet to a parsed XML document. But combining the two does not have anything to do with the claimed limitation.

The Examiner focuses on DOM Tutorial, page 3, reproduced below:

Application No. 10/026,364

Atty Docket No. JGR 1007-1

Accessing the DOM Tree

You can access the tree starting at the root and walking down the tree or by querying for a specific node. You navigate to the root element using the documentElement property which returns the root element as an XMLDOMNode object.

```
Dim xmlDoc As New DOMDocument
Dim root As IXMLDOMElement
Dim child As IXMLDOMNode

xmlDoc.load("reports.xml")

'Set root to the root element collection.
Set root = xmlDoc.documentElement

'Walk from the root to each of its child nodes.
For Each child In root.childNodes
```

The Examiner reasons, "reports.xml" is loaded. The loading implies that 'reports.xml' is within the current working directory." This reasoning does not connect with claim 4. There is no mention of a document ID or even more than one document in the cited passage of DOM Tutorial. The claim does not mention a current working directory. The reasoning does not make out a *prima facie* case under § 103(a).

In general, adding DOM Tutorial to the references cited does not change the way that XSLT works, because XSLT often processes an XLM document that has been parsed into a DOM tree. There is no teaching, suggestion or motivation to introduce a different combination of XSLT and DOM than is conventional.

Therefore, claim 4 should be allowable over Harold in view of Yalcinalp in further view of DOM Tutorial.

Claims 5 and 8

Claims 5 and 8 should be allowable over Harold in view of Yalcinalp in further view of DOM Tutorial for at least the same reasons as claim 1, from which they depend.

Claim 17

Claim 17 includes the limitations:

further including accessing an element set list corresponding to a plurality of the update values to be applied at the path specification.

These limitations are not found in Harold in view of Yalcinalp in further view of DOM Tutorial. Our reasoning is set forth above, in the context of claims 15-16.

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

Therefore, claim 17 should be allowable over Harold in view of Yalcinalp in further view of DOM Tutorial for at least the same reasons as claims 15-16, to which claim 17 is analogous.

Applicants respectfully submit that claims 4-5, 8 and 17 should be allowable over Harold in view of Yalcinalp in further view of DOM Tutorial.

Rejection Under 35 U.S.C. § 103(a) of Claims 9-10, 12-13, 18-19, and 21-22

The Examiner rejects **claims 9-10, 12-13, 18-19, and 21-22** under 35 U.S.C. § 103(a) as unpatentable over Harold, XML in a Nutshell, January 2001 in view of Yalcinalp (US 6,507,857) in further view of Oguji, Validating XML with Schematron.

Claims 9-11

Claims 9-11 include the limitations:

wherein the self-describing, structured document includes a document type, further including accessing a schema corresponding to the document type and validating application of the update operator and the update values at the path specification

These claims depend, respectively, from claims 1, 7 and 8. The common limitations are not found in Harold in view of Yalcinalp in further view of Oguji.

First, the cited references do not update an XML document. Second, Schematron in Oguji is not applied after transformation of the XML document.

In general, adding Oguji's variation on Schematron to the references cited does not change the way that XSLT works. Schematron is illustrated as validating an XML document, for instance, when it is parsed into a DOM tree. XSLT often processes an XML document after it has been validated and parsed into a DOM tree. There is no teaching, suggestion or motivation to introduce a different combination of XSLT, DOM and Schematron than is conventional.

Therefore, claims 9-11 should be allowable over Harold in view of Yalcinalp in further view of Oguji.

Claims 12-14

Claims 12-14 include the limitations:

[9] wherein the self-describing, structured document includes a document type, further including accessing a schema corresponding to the document type and validating application of the update operator and the update values

Application No. 10/026,364

Atty Docket No. JGR 1007-1

*at the path specification**[12-14] wherein the schema is compliant with any version of a SOX standard*

The limitations common to claims 12-14 are not found in Harold in view of Yalcinalp in further view of Ogbuji.

The Examiner says, "Ogbuji further discloses a SOX standard", which is true. However, there is no teaching suggestion or motivation to combine the references to produce the claimed invention, except the claims. This is one place where avoiding hindsight is important.

The standard for evidence of a teaching, suggestion or motivation to combine was reset by *In re Lee*, 277 F.3d 1338, 1342-44, 61 USPQ2d 1430, 1433-34 (Fed. Cir. 2002). The Federal Circuit clarified the need for evidentiary quality support of an examiner's factual basis for finding a teaching, suggestion or motivation in the prior art (as opposed to an examiner's opinion), 277 F.3d at 1343-44:

As applied to the determination of patentability *vel non* when the issue is obviousness, "it is fundamental that rejections under 35 U.S.C. § 103 must be based on evidence comprehended by the language of that section." *In re Grasselli*, 713 F.2d 731, 739, 218 U.S.P.Q. (BNA) 769, 775 (Fed. Cir. 1983). ... "The factual inquiry whether to combine references must be thorough and searching." *Id.* It must be based on objective evidence of record. This precedent has been reinforced in myriad decisions, and cannot be dispensed with. [citation omitted] The need for specificity pervades this authority. See, e.g., *In re Kotzab*, 217 F.3d 1365, 1371, 55 U.S.P.Q.2D (BNA) 1313, 1317 (Fed. Cir. 2000) ("**particular findings must be made as to the reason the skilled artisan, with no knowledge of the claimed invention, would have selected these components for combination in the manner claimed**"); *In re Rouffet*, 149 F.3d 1350, 1359, 47 U.S.P.Q.2D (BNA) 1453, 1459 (Fed. Cir. 1998) ("even when the level of skill in the art is high, the Board must identify specifically the principle, known to one of ordinary skill, that suggests the claimed combination. In other words, the Board **must explain the reasons one of ordinary skill in the art would have been motivated to select the references and to combine them to render the claimed invention obvious.**"); *In re Fritch*, 972 F.2d 1260, 1265, 23U.S.P.Q.2D (BNA) 1780, 1783 (Fed. Cir. 1992) (the examiner can satisfy the burden of showing obviousness of the combination "only by showing **some objective teaching** in the prior art or that knowledge generally available to one of ordinary skill in the art **would lead that individual to combine the relevant teachings of the references**"). ... In its

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

decision on Lee's patent application, the Board rejected the need for "any specific hint or suggestion in a particular reference" to support the combination of the Nortrup and Thunderchopper references. Omission of a relevant factor required by precedent is both legal error and arbitrary agency action.

(Emphasis added.) The Examiner's reasoning does not acknowledge or satisfy this standard.

The Examiner's argument relies on hindsight, because part of the motivation (to combine the references in the manner claimed) is coming from the Examiner, not from any of the references. All that Ogbuji discloses is that SOX is a schema standard. When the Examiner says to combine the references as claimed, the Examiner is motivating the claimed combination, not Ogbuji. The only source in this case for teaching how to combine the references is Applicants' claims. Therefore, the Examiner improperly relies on hindsight, which is difficult to avoid when the references do not teach or suggest the claimed combination. 2-5 Chisum on Patents § 5.03 [2][c] n. 29 (2005 Lexis version); *e.g. ATD Corp. v. Lydall, Inc.*, 159 F.3d 534, 546, 48 USPQ2d 1321, 1329 (Fed. Cir. 1998) ("Determination of obviousness can not be based on the hindsight combination of components selectively culled from the prior art to fit the parameters of the patented invention."); *Grain Processing Corp. v. American Maize-Products Corp.*, 840 F.2d 902, 907, 5 USPQ2d 1788, 1792 (Fed. Cir. 1988) ("Care must be taken to avoid hindsight reconstruction by using 'the patent in suit as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit.'").

Therefore, claims 12-14 should be allowable over Harold in view of Yalcinalp in further view of Ogbuji.

Claims 18-20

Claims 18-20 are similar to 12-14. They include the limitations:

further including accessing a set of business processing rules corresponding to the document type and validating application of the update operator and the update values at the path specification

The claims, respectively, depend from claims 9, 10 and 11. The common limitations are not found in Harold in view of Yalcinalp in further view of Ogbuji, because there is no teaching, suggestion or motivation to combine the three references cited in the

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

manner claimed. The only source for reconstructing the claimed method is the claims themselves.

Therefore, claims 18-20 should be allowable over Harold in view of Yalcinalp in further view of Ogbuji.

Claims 21-23

Claims 21-23 include the limitations:

wherein the business processing rules are Schematron-compliant

These limitations are not found in Harold in view of Yalcinalp in further view of Ogbuji.

Claims 21-23 should be allowable over Harold in view of Yalcinalp in further view of Ogbuji for at least the same reasons as claims 18-21, from which they depend.

Applicants respectfully submit that claims 9-10, 12-13, 18-19, and 21-22 should be allowable over Harold in view of Yalcinalp in further view of Ogbuji. In addition, claims 11, 14, 20 and 23 should be allowable for the reasons given.

Rejection Under 35 U.S.C. § 103(a) of Claims 11, 14, 20 and 23

The Examiner rejects **claims 11, 14, 20 and 23** under 35 U.S.C. § 103(a) as unpatentable over Harold, in view of Yalcinalp in further view of Document Object Model (DOM) Tutorial and of Oguji, Validating XML with Schematron.

We addressed these claims in the context of claims 9-10, 12-13, 18-19 and 21-22, immediately above.

Applicants respectfully submit that claims 11, 14, 20 and 23 should be allowable over Harold in view of Yalcinalp in further view of DOM Tutorial and of Ogbuji.

Rejection Under 35 U.S.C. § 103(a) of Claims 30-40

The Examiner rejects **claims 30-37** under 35 U.S.C. § 103(a) as unpatentable over Harold, in view DOM Tutorial.

The Examiner rejects **claims 38 & 40** under 35 U.S.C. § 103(a) as unpatentable over Harold, in view DOM Tutorial in further view of Ogbuji, Validating XML with Schematron.

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

The Examiner rejects **claim 39** under 35 U.S.C. § 103(a) as unpatentable over Harold, in view DOM Tutorial in further view of Ogbuji and "XML and EDI: Peaceful Co-Existence" (EDI).

Claim 30

Claim 30 includes the limitations:

receiving a request identifying a starting document and specifying a document type to be generated from the starting document;

accessing at least first and second declarative transformations corresponding to the starting document and the specified document type;

applying the first declarative transformation to the starting document, producing a resulting document of the specified document type;

applying the second declarative transformation to the resulting document, producing character string data including a plurality of

path specifications for nodes in the resulting document;

starting values copied from the starting document for at least some of the nodes; and

editable values for at least some of the nodes;

responding to the request with the character string data;

receiving an updated version of the character string data; and

producing an updated resulting document corresponding to the updated version of the character string data.

These limitations are not found in Harold in view of DOM Tutorial.

The Examiner is urged to read this claim carefully, as it is very involved. One begins with a starting document, such as a request for quotation (see [0053]), and generates a draft resulting document. Then, from the draft resulting document, one generates a form ([0054]-[0056]) to display to a user for editing. The user input is applied to the draft resulting document, matching it to the user's requirements, for instance, to produce a quotation in response to a request for quotation.

The Examiner's analysis at OA 10-11 does not attempt to arrange elements in the claimed combination. For instance, there is no citation to a reference that includes both first **and** second declarative transformations. There is no citation to a reference

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

that proceeds in three phases, applying user input after first and second declarative transformations.

There is no suggestion, teaching or motivation in any of the cited references to produce the claimed combination. Harold admittedly lacks many elements. The DOM Tutorial, p. 2, "Loading and Saving" merely describes building a DOM tree from an XML document and saving the DOM tree:

Loading and Saving Data

Use the Load or LoadXML methods to load an XML file into the DOM. The load method uses a path or url to an XML file. The loadXML method loads a string containing the XML data. After loading, XMLDoc, contains a tree consisting of the parsed contents of reports.xml.

```
xmlDoc.load("http://xmlfiles/reports.xml")
xmlDoc.load("c:\\temp\\reports.xml")

xmlDoc.loadXML("<customer><first_name>Joe</first_name>
<last_name>Smith</last_name></customer>")
```

To save a parsed XML document to a file use the Save method. Save can take a file name as a string.

```
xmlDoc.save("c:\\temp\\reports.xml")
```

This cited passage does not suggested the claimed method *as a whole*.

The Examiner's proposed motivation is, "It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have combined Harold's method with Document's [DOM Tutorial's] method, since it would have allowed the user to interface with the structure document (Document [DOM Tutorial]: page 1, "What is a DOM?" paragraph 2)." This motivation does not connect with claim 30, as a whole. That is, the motivation fails to motivate the skilled artisan, with no knowledge of the claimed invention, to select the claimed components for combination in the manner claimed. See, *In re Lee, supra*, 277 F.3d at 1342-44; *In re Kotzab, supra*, 217 F.3d at 1371. Therefore, the proposed motivation is incomplete and not sufficient to make out a *prima facie* case.

The proposed motivation also fails because it is an exercise in hindsight. The DOM Tutorial reference, p. 1, ¶ 2 does not suggested the claimed invention *as a whole*.

Application No. 10/026,364**Atty Docket No. JGR 1007-1**

(Nor does the Examiner's reasoning, as explained above.) The only suggestion for producing the claimed combination, as a whole, is this patent application.

Therefore, claim 30 should be allowable over Harold in view of DOM Tutorial.

Claims 31-42

Claims 31-42 include limitations discussed above in the context of claims 1-29. These claims should be allowable for at least the same reasons as claim 30 and the claims to which they are analogous. They also should be allowable for failure to satisfy the *In re Lee* standard for an evidentiary quality showing of a teaching, suggestion or motivation to combine references.

Claims 31-33 are analogous to claim 2 and should be allowable for at least the same reasons.

Claim 34 is analogous to claim 5 and should be allowable for at least the same reasons.

Claims 35-37 are analogous to claim 6 and should be allowable for at least the same reasons.

Claim 38 is analogous to claim 11 and should be allowable for at least the same reasons.

Claim 39 is analogous to claim 18 and should be allowable for at least the same reasons.

Claim 40-41 are analogous to claim 12 and should be allowable for at least the same reasons.

Claim 42 is analogous to claim 21 and should be allowable for at least the same reasons.

Applicants respectfully submit that claims 30-42 should be allowable over Harold in view of DOM Tutorial and other references.

Rejection Under 35 U.S.C. § 103(a) of Claims 43-44

The Examiner rejects **claim 43** under 35 U.S.C. § 103(a) as unpatentable over Harold and Yalcinalp in further view of Fox (US 2002/011421).

Application No. 10/026,364

Atty Docket No. JGR 1007-1

The Examiner rejects **claim 44** under 35 U.S.C. § 103(a) as unpatentable over Harold and DOM Tutorial in further view of Fox.

Common to claims 43-44 are the limitations:

further including: displaying for a user a graphical interface adapted to interact with the user and generate the character string including the one or more sets

These limitations are not supplied by Fox.

From Fox, the Examiner cites [0192]-[0212]:

"[0192] display GUI, with links to functions, including:

"[0193] back (or previous) page;

"[0194] left and right page commands;

"[0195] next page;

"[0196] view options;

"[0197] file save options;

"[0198] index;

"[0199] table of contents;

"[0200] chapter/location selector;

"[0201] go to;

"[0202] find/search; and

"[0203] collaborate online; and

"[0204] enable audio/video decode functions, which includes:

"[0205] file source/embedded data files;

"[0206] and online source/streaming video and audio files;

"[0207] an enable multiple page viewers function;

"[0208] an enable page curl with text capture and hyperlink extensions functions;

"[0209] an enable IP/browser function;

"[0210] enable orthogonal search and data collection functions;

"[0211] enable text block autosummarize functions;

Application No. 10/026,364

Atty Docket No. JGR 1007-1

"[0212] and an open program function."

(Indentation of [0207]-[0212] corrected from patent, based on context.) In all likelihood, the Examiner only meant to cite [0192]-[0203], as audio/video decode functions are not mentioned in claims 43-44.

The list of Fox's GUI features in [0192]-[0203] does not read on, teach or suggest *a graphical interface adapted to ... generate the character string including the one or more sets*. A particular type of character string is claimed, by the antecedent basis of "the one or more sets". None of the references teach or suggest the claimed type of character string and Fox does not suggest a GUI interface adapted to generate the claimed type of character string.

Applicants respectfully submit that claims 43-44 should be allowable over Harold and Yalcinalp/DOM Tutorial in further view of Fox.

CONCLUSION

Applicants respectfully submit that the pending claims are now in condition for allowance and thereby solicit acceptance of the claims, in light of these amendments.

The undersigned can ordinarily be reached at his office at (650) 712-0340 from 8:30 a.m. to 5:30 p.m. PST, Monday through Friday, and can be reached at his cell phone at (415) 902-6112 most other times.

Respectfully submitted,



Dated: 24 October 2005

Ernest J. Beffel, Jr., Reg. No. 43,489

HAYNES BEFFEL & WOLFELD LLP
P.O. Box 366
Half Moon Bay, CA 94019
Telephone: (650) 712-0340
Facsimile: (650) 712-0263